



## INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification: <b>H04L 9/32</b>	A1	(11) International Publication Number: <b>WO 00/54458</b>
		(43) International Publication Date: <b>14 September 2000 (14.09.2000)</b>
(21) International Application Number: <b>PCT/US00/06313</b>		
(22) International Filing Date: <b>10 March 2000 (10.03.2000)</b>	<b>Published</b>	
(30) Priority Data: <b>09/268,084 12 March 1999 (12.03.1999) US</b>		
(60) Parent Application or Grant <b>PSIONIC SOFTWARE, INC. [/]; O. ROWLAND, Craig [/]; O. ROWLAND, Craig [/]; O. TAYLOR RUSSELL, Gail ; O.</b>		

(54) Title: INTRUSION DETECTION SYSTEM

(54) Titre: SYSTEME DE DETECTION D'INTRUSION

## (57) Abstract

A computer-implemented intrusion detection system and method (1) that monitors a computer system in real-time for activity indicative of attempted or actual access by unauthorized persons or computers. The system detects unauthorized users (20) attempting to enter into a computer system by comparing user behavior to a user profile (22), detects events that indicate an unauthorized entry into the computer system (90), notifies a control (37, 97) function about the unauthorized users and events that indicate unauthorized entry into the computer system and has a control function (125) that automatically takes action in response to the event (127). The user profiles are dynamically constructed for each computer user when the computer user first attempts to log into the computer system (24) and upon subsequent logins (25), the user's profile is dynamically updated (25). By comparing user behavior to the dynamically built user profile (3-5), false alarms are reduced. The system also includes a log auditing function (10), a port scan detector (75) and a session monitor function (90).

## (57) Abrégé

L'invention concerne un procédé (1) et un système de détection d'intrusion informatisé, qui surveille un système informatique en temps réel et signale une tentative d'accès ou un accès réel par des personnes ou des ordinateurs non autorisés. Le système détecte les utilisateurs non autorisés (20) qui tentent d'entrer dans un système informatique par comparaison d'un comportement d'utilisateur avec un profil d'utilisateur (22), détecte les événements indiquant une entrée non autorisée dans le système informatique (90), notifie une fonction de contrôle (37, 97) stockant des informations associées aux utilisateurs et aux événements non autorisés indiquant une entrée non autorisée dans le système informatique, et exécute une fonction de contrôle (a25) qui réagit automatiquement à l'événement (127). Les profils d'utilisateur sont construits en dynamique pour chaque utilisateur d'ordinateur lorsque ledit utilisateur tente, pour la première fois, d'entrer en communication avec le système informatique (24), et ces profils sont mis à jour (25) en dynamique lors d'entrées en communication (25) ultérieures. Les fausses alertes sont réduites par comparaison du comportement d'utilisateur avec le profil d'utilisateur (3-5) construit en dynamique. Le système comprend également une fonction de vérification (10) d'entrée en communication, un détecteur de balayage (75) de port, et une fonction de surveillance (90) de session.

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION  
International Bureau

## INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 :  H04L 9/32		A1	(11) International Publication Number: WO 00/54458  (43) International Publication Date: 14 September 2000 (14.09.00)
(21) International Application Number: PCT/US00/06313 (22) International Filing Date: 10 March 2000 (10.03.00)  (30) Priority Data: 09/268,084 12 March 1999 (12.03.99) US		(81) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).	
(71) Applicant (for all designated States except US): PSIONIC SOFTWARE, INC. [US/US]; 6908 Dogwood Hollow, Austin, TX 78750 (US).  (72) Inventor; and (75) Inventor/Applicant (for US only): ROWLAND, Craig [US/US]; 6908 Dogwood Hollow, Austin, TX 78750 (US).		Published <i>With international search report.          Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.</i>	
(74) Agent: TAYLOR RUSSELL, Gail; Taylor Russell & Russell, P.C., Building One, Suite 1200, 4807 Spicewood Springs Road, Austin, TX 78759 (US).			
(54) Title: INTRUSION DETECTION SYSTEM			
(57) Abstract			
<p>A computer-implemented intrusion detection system and method (1) that monitors a computer system in real-time for activity indicative of attempted or actual access by unauthorized persons or computers. The system detects unauthorized users (20) attempting to enter into a computer system by comparing user behavior to a user profile (22), detects events that indicate an unauthorized entry into the computer system (90), notifies a control (37, 97) function about the unauthorized users and events that indicate unauthorized entry into the computer system and has a control function (125) that automatically takes action in response to the event (127). The user profiles are dynamically constructed for each computer user when the computer user first attempts to log into the computer system (24) and upon subsequent logins (25), the user's profile is dynamically updated (25). By comparing user behavior to the dynamically built user profile (3-5), false alarms are reduced. The system also includes a log auditing function (10), a port scan detector (75) and a session monitor function (90).</p>			

**FOR THE PURPOSES OF INFORMATION ONLY**

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece	MN	Mali	TR	Turkey
BG	Bulgaria	HU	Hungary	MR	Mongolia	TT	Trinidad and Tobago
BJ	Benin	IB	Ireland	MW	Mauritania	UA	Ukraine
BR	Brazil	IL	Israel	MW	Malawi	UG	Uganda
BY	Belarus	IS	Iceland	MX	Mexico	US	United States of America
CA	Canada	IT	Italy	NE	Niger	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NL	Netherlands	VN	Viet Nam
CG	Congo	KE	Kenya	NO	Norway	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NZ	New Zealand	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	PL	Poland		
CM	Cameroon	KR	Republic of Korea	PT	Portugal		
CN	China	KZ	Kazakhstan	RO	Romania		
CU	Cuba	LC	Saint Lucia	RU	Russian Federation		
CZ	Czech Republic	LI	Liechtenstein	SD	Sudan		
DE	Germany	LK	Sri Lanka	SB	Sweden		
DK	Denmark	LR	Liberia	SG	Singapore		

**Description**

5

10

15

20

25

30

35

40

45

50

55

5

**INTRUSION DETECTION SYSTEM**  
**TECHNICAL FIELD OF THE INVENTION**

10        5        The present invention relates generally to intrusion detection for a computer system. More particularly, the invention is a computer-implemented intrusion detection system and method that monitors a computer system for activity indicative of attempted or actual access by unauthorized persons or computers.

15

10        10      Because of the increasing reliance on Internet, Intranet and extranet network computer access, intrusion into computer systems by unauthorized users is a growing problem. An intrusion is unauthorized access or attempted access into or unauthorized activity in a computer or information system. Intrusion detection technologies are therefore becoming extremely important to improve the overall security of computer systems.

20

15        15      Intrusion detection is the process of identifying that an intrusion has been attempted, is occurring or has occurred.

25

20        20      In most intrusion detection systems, data may be automatically collected and reduced but the analysis of that data usually remains manual. Profiling and pattern recognition techniques also have been used to analyze the data collected and presented to an intrusion detection system. The off-line analysis involves determining normal behavior for a user, application or system. The normal behavior is then used to develop sets of rules. Significant deviations from the rules, referred to as anomalous behavior, may then be flagged as potential intrusions. Some intrusion detection systems, based on anomaly detection techniques, look for statistically anomalous behavior, that is, behavior that

30

25        35      appears unusual when compared to other user behavior. One drawback of anomaly detection systems is that they are prone to both false positive and false negative alerts because the rules are general in nature and not specific for the behavior of each user. False positives occur when the intrusion detection system identifies an event as an

40

40        30      intrusion when none has occurred. False positives may divert the attention and time of the

45

30        45      system administrator and security staff and if frequent enough, may cause a lack of confidence in the intrusion detection system. False negatives are instances where the intrusion detection system fails to detect an intrusion while it is occurring or after it has occurred. The result may be slow or no response to the intrusion that can result in financial loss and system damage. False negatives often occur because the models used to profile

35        50      the anomalous behavior do not adequately predict the intruder behavior and its result within the computer system.

50

55

Some intrusion detection systems use expert systems, which are driven from an  
5 encoded rule base to monitor policy compliance. The expert system applies the rules to  
assure all users are operating within their privileged rights. Even in the expert system, the  
encoded rules are usually generated by profiling the anomalous behavior and then building  
5 a rule based system. This means that the expert system intrusion detection system at  
10 present suffers from the same problems such as false positives and false negatives as the  
anomalous detection systems. Other systems have passive monitor functions that  
continually analyze data presented to them. They are similar to antivirus functions in that  
they can only detect what has been defined to them. Another type of intrusion detection  
15 10 system is a scanner. Unlike other intrusion detection tools that report when a threshold has  
been exceeded, scanners actively attempt to find security holes (called vulnerabilities) and  
unauthorized hardware and software.

The above mentioned intrusion detection methods have many drawbacks. One is  
20 that the systems can only detect and monitor what has been previously defined to them,  
15 either using expert system rules or rules developed through data collection reduction and  
analysis or through profiling. This can result in false negatives because unknown attacks  
have not been previously defined. In addition, most systems only analyze and develop  
25 profiles and patterns after the fact. These profiles and patterns of behavior are  
subsequently incorporated into rule-based systems to recognize future attacks. Even in  
those instances where alerts are issued in near real-time, valuable time and the intruder's  
30 trail can be lost. In addition, many of these systems require human intervention, both in the  
initial analysis of data and profile and pattern recognition building steps and when an  
anomalous event has occurred, to determine the action to be taken. Relying on human  
35 intervention can delay the identification of the intrusion and may not prevent network  
damage or exploitation.

To be able to detect intrusions as they are occurring or soon after, there is a need  
40 for the intrusion detection system to be a real-time system. There is a need to automatically  
build profiling data specific for each user or class of users that can be used to determine  
normal actions for a user to reduce the occurrence of false alarms and to improve  
45 detection. There is a need for a system that can detect suspicious actions, determine the  
source and institute autonomous responses. There is also a need for the intrusion  
detection system to take automatic action, without waiting for a human administrator to  
intervene and act, to mitigate the effects of an intrusion and to prevent future actions.  
There is also a need to coordinate information transfer within host, multi-host and network  
35 environments so responses to intrusions can be coordinated. In addition, there is a need to

5

combine the above listed capabilities with real-time monitoring of log audit files, port scan detection capability and session monitoring.

10

The present invention is a computer implemented method for detecting intruders in a computer system. The method comprising the steps of detecting an unauthorized user attempting to enter into a computer system by comparing actions of the user to a dynamically built profile for the user, and if the action is out of range of the user profile, notifying a control function. If events are detected that indicate an unauthorized entry into the computer system has occurred by comparing and if an event occurs that indicates unauthorized entry, a control function is notified, and automatically executes a specific action in response to the event.

15

The dynamically built user profile comprises dynamically constructing a user profile for each computer user when the computer user first attempts to log into the computer system, dynamically updating the user profile for the user for each attempt by the user to log into the system after the first attempt, and updating the user profile when the user logs out of the computer system.

20

Dynamically monitoring computer system log files comprises monitoring for events that indicate an unauthorized attempted entry into the computer system. Dynamically

25

monitoring system log files comprises comparing the system log files to events to ignore and ignoring the event if the system log file indicates a match with the event to ignore and comparing the system log files to events known to indicate an unauthorized entry into the computer system and notifying a control function about the unauthorized entry and automatically executing a specific action in response to the event by the control function.

30

The method further comprises dynamically monitoring user actions after the user has logged into a computer system for unauthorized access by the user to system information, and if unauthorized access occurs, notifying a control function about the unauthorized access and automatically executing a specific action in response to the event by the control function. The method dynamically monitors user actions after the user has logged into a computer system for corruption of system information by the user and if corruption of system information occurs, a control function is notified and automatically executes a specific action in response by the control function.

40

The method further comprises scanning network ports to determine if a user has connected to more than a selected number of network ports. If the user has exceeded the selected number of network ports, the control function is notified and automatically executes a specific action in response to the. The selected number of network ports may be set by the system administrator.

45

50

55

5           The detecting events that indicate an unauthorized entry into the computer system comprise detecting anomalous events when a user logs out of the computer system. This comprises monitoring a user's file history to determine if the user's file history has been altered, monitoring computer system files to determine if a modification has been made  
5           10       that indicates an unauthorized intrusion into the computer system, monitoring a user's computer files to determine if a modification has been made that indicates an unauthorized intrusion into the computer system, determining if a program has been left running that should have stopped running when the user logs out of the computer system. If an anomalous event has been a control function is notified about the anomalous event and the  
15           10       control function takes user specified action in response to the anomalous event.

20           Detecting unauthorized users comprises if the user has attempted to log in from a computer host that is not allowed access to the computer system, notifying a control function about the attempted login and allowing the control function to take a user specified action in response. Detecting unauthorized users also comprises if the user attempts to log  
25           15       into the computer system and has an active login, checking to determine if the user is allowed to have more than one login active simultaneously, and if not, notifying a control function about the attempted login and automatically executing a specific action in response to the event by the control function.

25           The dynamically constructed user profile for each computer user comprises storing user name, login terminal, time of creation of initial user profile, time of user's first login, time history of the user's logins, time periods that the user is allowed to log into the system and total number of logins when the computer user. The user profile may be stored in a user profile database.

30           Dynamically updating the user profile for the user comprises, for each user, entering the current login time, login terminal, updating the time history of the user's login and incrementing the total number of logins.

35           40       Detecting unauthorized users comprises if the user has attempted to log in at a time different from the time periods that the user is allowed to log into the computer system, notifying a control function about the attempted login and allowing the control function to  
30       35       take user specified action in response.

45           45       The method further comprises dynamically constructing a list of active users logged into the computer system and dynamically updating the list of active users when a user logs into the system and logs out of the system. The list of active users may comprise user name, user terminal and time of user login.  
50           35       The control function comprises storing information about unauthorized users and events that indicate an unauthorized entry into the computer system, taking action in response to

5 the unauthorized users and events. The action is selected from the group consisting of  
logging the information in a local controller, sending the information to a network controller,  
disabling the unauthorized user's account,

10 5 blocking access to the computer system for the user, notifying a system administrator and  
ignoring the unauthorized user and unauthorized entry. The action taken may be defined  
by the system administrator prior to initialization of the intrusion detection system.

15 10 The control function may be located in a local computer where the unauthorized  
user and unauthorized entry occurred. The control function in the local computer sends  
information about unauthorized users and events to a central computer connected to the  
local computer. Alternatively, the control function may be located in a central computer  
connected to the local computer. Multiple local computers may be connected to the central  
computer.

20 20 The central computer comprises performing centralized analysis of unauthorized  
users and events, performing correlation of unauthorized users and events from the  
multiple local computers, alerting a central computer system administrator, and sending the  
analysis and correlation results to the multiple local computers.

25 25 The method further comprises, for each user, continuously monitoring user activity  
for a threat to the computer system. Continuously monitoring comprises analyzing user  
command entries and comparing the entries to known threat events and known attack  
20 30 patterns indicating a computer intrusion and if a match occurs, notifying the control function  
and allowing the control function to take user specified action in response. Continuously  
monitoring the system process accounting records comprises comparing the entries to  
known threat events and known attack patterns indicating a computer intrusion and if a  
match occurs, notifying the control function and allowing the control function to take user  
35 35 specified action in response.

40 40 The method further comprises continuously monitoring commands entered by the  
user and comparing the commands to known threat events and known attack patterns  
indicating a computer intrusion and if a match occurs, notifying the control function and  
allowing the control function to take user specified action in response. The method further  
30 45 comprises continuously monitoring network port activity and comparing the activity to  
known threat events and known attack patterns indicating a computer intrusion and if a  
match occurs, notifying the control function and allowing the control function to take user  
specified action in response. The action taken may be selected from the group consisting  
of logging the event, removing the user from the computer system and executing a  
35 50 selected command.

5                   The computer-implemented methods are embodied in software programs that may  
be stored on a computer-readable medium.

#### SUMMARY

The present invention provides a real-time intrusion detection method and system.

10                 5   The intrusion detection system automatically and dynamically builds user profile data  
(known as a signature) for each user (or alternatively, a class of users) that can be used to  
determine normal actions for each user to reduce the occurrence of false alarms and to  
improve detection. The user profile data (signature) is saved and updated every time the  
user logs on and off the system. The advantage of dynamically building user profile data  
15                 10   based on past user behavior and comparing it to that user's current behavior is that the  
number of false alarms is reduced. In addition, there is no need to enter sets of rules prior  
to system initialization. The system detects suspicious actions, determines the source and  
institutes autonomous responses. The system acts to mitigate the effects of an intrusion  
20                 15   and to prevent future actions without waiting for human action. The automatic actions to be  
taken can be specified by the system administrator prior to initialization of the system. The  
automatic actions can be tailored to address the specific anomaly detected by the intrusion  
25                 20   detection system. For example, through a local or system controller, the system can log the  
events, disable user accounts and block access to the system. In one embodiment, the  
system coordinates information transfer within host, multi-host and network environments  
30                 25   to coordinate intrusion response. The system combines the above listed capabilities with  
real-time monitoring of log audit files, port scan detection capability and session monitoring.  
Throughout this document, use of the terms dynamic or dynamically in relation to a process  
means that the process is operating in real-time or close to real-time.

#### BRIEF DESCRIPTION OF THE DRAWINGS

35                 35   These and other features, aspects and advantages of the present invention will  
become better understood with regard to the following description, appended claims and  
accompanying drawings where:  
40                 40   Fig. 1 shows a functional block diagram of the host based intrusion detection  
system.  
45                 30   Fig. 2 is a block diagram of the log file auditing function.  
                    35   Fig. 3 is a flow diagram of the login anomaly detection function.  
                    45   Fig. 4 is a flow diagram of the user profile database and active user database  
update function.  
                    50   Figs. 5A and 5B are flow diagrams of the logout anomaly detection function.  
                    35   Fig. 6 is a flow diagram of the port scan detector function.  
                    55   Fig. 7 is a flow diagram of the session monitor function.

5 Fig. 8 is a flow diagram of the controller function.

Fig. 9 is a block diagram of an alternate embodiment of a host based intrusion  
detection system having a central system controller.

Fig. 10 is a flow diagram of program setup for the intrusion detection systems.

10 5 DETAILED DESCRIPTION OF THE DRAWINGS

Fig. 1 shows a functional block diagram of the intrusion detection system. The  
system is comprised of a log audit function 2, a login anomaly detection function 3, a logout  
anomaly detection 7, a session monitor function 4 and a port scan detector function 5  
interfacing with a local controller function 6. The log audit function 2, login anomaly

15 10 detection function 3, logout anomaly detection function 7, session monitoring function 4  
and port scan detector function 5 all operate in real-time to detect activity indicative of an  
attack by unauthorized users or systems. The log audit function continuously monitors  
system log files for anomalous activity which can include known suspicious activity and  
unknown system anomalies. When anomalous behavior occurs, the log audit function 2

20 15 notifies the controller 6 and sends information about the activity to the controller 6 for  
further processing. The log auditing function 2 is described in Fig. 2. The login anomaly  
detection function 3 monitors system login activity and when anomalous behavior is  
detected, notifies the controller and sends information about the activity to the controller 6  
for further processing. The login anomaly detection function 3 is described in Fig. 3. The

25 20 logout anomaly detection function 7 monitors system logout activity and if anomalous  
behavior is detected, notifies the controller and sends information about the activity to the  
controller 6 for further processing. The logout anomaly detection function 7 is described in  
Figs. 5A and 5B. The session monitoring function 4 watches user activity after a login has  
been established. The function continuously watches keystrokes for known attack

30 25 signatures and suspicious activity. Signatures are kept in a user-editable database on the  
local machine. Once suspicious or known attack activity is detected, the session monitor 4  
will send information about the activity to the controller 6 for further processing. The  
session monitoring function 4 is described in Fig. 7. The port scan detector function 5

35 40 monitors Internet ports (such as TCP and UDP) for port scanning activity which is a method  
used by attackers to determine the vulnerabilities of a target host and to run a series of  
attacks to gain entry on the vulnerable target host. When the port scan detection function 5  
detects port scanning activity, it sends information about the activity to the controller for  
further processing. The port scan detector function 5 is described in Fig. 6. The controller

45 45 function 6 controls all actions that the host-based intrusion detection system may perform  
upon being notified from the log audit function 2, the login anomaly detection function 3, the  
logout anomaly detection function 7, the session monitor 4 or the port scan detector

5 function 5 that an anomalous activity has occurred, the controller takes an appropriate  
action based on that activity. The controller function is described in Fig. 8.

10 Turning now to Fig. 2, a block diagram of the log file auditing function is shown.  
The log auditing function 10 monitors the system login auditing files 11 by comparing the  
log file activity known attack events 12, known security violations 13, and events to ignore  
14. If the log file activity indicates a known attack event 12 or a known security violation 13  
indicating a suspicious event or unknown event has occurred or is in the process of  
occurring, then the log auditing function 10 constructs a message containing the log file  
information and signature identification information and forwards it to the controller for  
15 action. The log auditing function can run on a periodic basis with the period selected by  
the user or it can run continuously in real-time. The user has the flexibility to add or remove  
functions within the login anomaly detection to customize the system.

20 Turning now to Fig. 3, a flow diagram of the login anomaly detection function 20 is  
shown. The system monitors login and logout audit files and logs (records) all logins and  
25 logouts for the target host 21. The target host is the computer that the user is logging into  
or logging out of. The system login auditing files may be login records (such as wtmp and  
utmp records) for a Unix® based operating system or may be event logs for a Windows  
NT® operating system. The system checks to determine if the user should be ignored 38.  
Certain users are not checked for login or logout anomalies. If the user is to be ignored  
30 processing continues at step 35 where the user is logged into the system. If the user is not  
to be ignored and if the user is logging in to the system, the monitor builds/updates the  
user profile database 22 and updates the active user database as shown in Fig. 4. The  
system administrator has the flexibility to add or remove functions within the logout  
anomaly detection to customize the system.

35 Turning now to Fig. 4, a flow diagram is shown of the user profile database and  
active user database update function. If the user is not in the user profile database 23, then  
the user is a new user and process first login function is executed 24. A new user profile  
entry is created 24 which contains the user name, the login host, the login terminal  
40 (sometimes called the TTY), the time of creating the initial user profile, the time of the  
user's first login, the set days and hours the user is allowed login access, the version of the  
database record type and sets the initial number of logins to one. In addition, the system  
administrator notified whenever a user logs into a host for the first time. If the user is  
already in the user profile database 23, then a user profile entry already exists for this user  
45 and that profile is updated 25. The updates to the user profile include appending the login  
time, login host and incrementing the total number of logins. The system also checks to  
determine that the user's login account is still valid, that is that it has not been disabled by  
50

5 a system administrator. An entry is created in the active user database 36 which contains  
the user's name, the terminal the user is logged in on, the time of login for this entry and  
the version of the database record.

10 Turning back to Fig. 3, the next step is to check to determine if the login is from a  
5 foreign domain 26. A foreign domain is one that is not contained within or allowed access  
to the host where the login is attempted. The list of allowed domains within the system is  
accessed 27 and if the login domain is not listed, it is considered foreign and the control  
function is notified 37.

15 The user login is checked to determine if there are multiple concurrent logins for the  
10 same user 28. A multiple concurrent login means that a user is logged into the system more  
than once from one or more different hosts concurrently. This type of behavior may indicate  
an intrusion. The log file is checked to determine if a user is logged in from one or more  
20 different hosts concurrently. If so and the user is not allowed to have multiple logins 29,  
then this login entry is denied and the multiple users are logged off from the system 30.

25 15 The next step is to determine if the user is logged in at an unusual time 31. For  
each user, a profile is automatically built of the days, times and length of time that the user  
has logged in. Once a certain threshold number of user logins have occurred for this user  
to allow for accurate user profiling (usually approximately ten logins, but this can be  
adjusted by the user), the day and time of the current user's attempted login is compared to  
20 that profile. If the current login time differs from the user's login profile, the control function  
is notified 37.

30 30 The next step is to compare the login activity with known attack patterns 34. If the  
login activity is similar to a known attack pattern, then the control function is notified 37.  
Next the history file is checked for suspicious command entries 39.

35 25 If these steps are successfully completed, the user is logged in 35 and the user's  
profile database entry is updated and the active user database is updated to track the login  
state of the user.

40 40 Turning now to Figs. 5A and 5B, a flow diagram of the logout anomaly detection  
function is shown. When a user attempts to logout, the logout anomaly detector 49 goes  
30 through a series of steps to process the logout to determine if something has occurred  
during the user's login time that may indicate a system anomaly. The logout entry for the  
user is updated in the user profile and the active user database is updated 50. If the user is  
45 to be ignored 65, then no other checking is done and the user is successfully logged out  
70. The next step is to determine if the user's file history has been compromised 51. If the  
35 history file no longer exists 52, the history file has been truncated 53 or the history file is a  
symbolic link 54, the event is logged and information about the event is sent to the

5 controller 55. The system examines the rhost file and other system authentication files to determine if dangerous security modifications to the host file have occurred 56. For example, entering a wildcard symbol, allows the host to allow anyone to log in without a password. If the host has been altered to allow anyone to login without a password or if  
5 other activity has occurred that may compromise security, the event is logged and  
10 information about the event is sent to the controller. The next step is to determine if the user's home directory contains one or more suspicious directories 59. Intruders will sometimes name a local directory in an odd way to hide their work. The system checks for known suspicious directories 60 and if it finds any, it may log the event and send  
15 information about the event to the controller 55. If a network computer process (sometimes called a "daemon") is left operating after logout 63 this could indicate a suspicious login and event is logged and information about the event is sent to the controller 55. The system checks to determine if the system audit records have been altered or are missing  
20 66. If so, the control function is notified 55. Next the program checks an administrator  
25 generated list of generic files to see if one or more of them exists in the user's home directory 67. If so, the control function is notified. Next, if a suspicious directory name is found 68, the control function is notified 55. If an rhost file exists, the control function is notified 69.

Turning now to Fig. 6, a flow diagram of the port scan detector is shown. Port scanning is a method used by attackers to determine the vulnerabilities of a target host. Once vulnerabilities are found, a series of attacks are usually run to gain entry. Port scanning makes use of the TCP/IP protocol, which is the core communication protocol of the Internet. It allows machines to communicate throughout the world in a reliable manner. One of its features is the use of protocol "ports" on remote and originating systems to establish connections between hosts. The ports available on a host are usually between the ranges of 1 to 65535, with ports 1 to 1024 being what is commonly referred to as "reserved" for use by critical Internet services. Each port that presents a service to a remote user is usually registered with the Internet Assigned Numbers Authority registry (IANA). This registration ensures that programs know what ports to avoid or specifically connect to depending on the services being requested. Examples of commonly used ports are:

- 21 – File Transfer Protocol (FTP) services.
- 25 – Simple Mail Transfer Protocol (SMTP) services.
- 80 – HTTP services (WWW servers)

35 When an attacker is looking for a new host to penetrate, they will often begin by looking for Internet programs that have known exploitable problems. These programs (called

5 "daemons") vary in number and degree of susceptibility to problems. As new problems are  
found the hacker community quickly makes use of them to penetrate more hosts. To  
facilitate looking for new victims, the attacker will use a program that may either: connect to  
all ports on the remote machine or deliberately pick one or more ports to search for a  
5 10 particular problem. Some of the ports may not answer, in which case the attacker moves  
on. Other ports will answer and the attacker can then glimpse at what problems they can  
take advantage of. Often attackers will go from host to host on the Internet looking for the  
same problem to exploit. An example port scan of a host may return the following  
information:

15 10       localhost     telnet   23/tcp  
             localhost     smtp     25/tcp  
             localhost     finger   79/tcp  
             localhost     http     80/tcp  
20       localhost     pop     110/tcp  
             localhost     imap     143/tcp

25 The port scan detector of the present invention alerts administrators that a person is  
actively looking for services on their host in a manner that indicates a hostile action. In the  
above port scan example our detector could present "fake" ports that an attacker will likely  
scan for. This could change the above port scan into the following:

30 20       localhost     fake     23/tcp (Fake port)  
             localhost     smtp     25/tcp  
             localhost     fake     79/tcp (Fake port)  
             localhost     http     80/tcp  
             localhost     fake     110/tcp (Fake port)  
35       localhost     fake     143/tcp (Fake port)

40 So even though in our example system there are only ports 25 and 80 active, the other  
ports will be tripwired by the port scan detector waiting for an attacker to unwittingly try to  
connect to them. When this occurs, the administrator or program can then take action to  
45 30 prevent this activity. In Figure 6, a flowchart of the port scan detector function 75 is shown.  
Internet ports (such as TCP and UCP) are monitored 76. If the port is in a list indicating  
that the port is not to be monitored 77, processing ends and no action is taken 78. If the  
port is in a list indicating it is to be monitored 77, the next step is to determine if the port is  
being used locally 79. If the port is being used locally it is temporarily removed from the  
35 45 monitored list until it is no longer used locally 80. If the port is not being used locally, the  
port is placed in the list of ports to be monitored 81. If the terminal or host computer where  
50

5           the user is attempting to log in from is a terminal or host to be ignored for port scanning 82,  
no action is taken 78. If the terminal or host computer is not to be ignored 82, then if the  
number of ports that are being scanned is less than a minimum number of ports 83, no  
action is taken 78. If the number of ports that are potentially being scanned is greater than  
5           10           the minimum number of ports 83, the next step is to determine if the terminal or  
host computer where the user is attempting to log in from is already blocked from the  
system 84. If so, no action is taken 78. If not, information about the apparent port scan is  
sent to the controller 85 and the controller then takes the appropriate action as discussed  
15           in Figs. 8 and 9 below. The appropriate actions can vary from logging the event, blocking  
10           access to the computer system from the attacking host or executing a user-supplied  
command.

20           Turning now to Fig. 7, a flow diagram of the session monitoring function 90 is  
shown. For each user, the session monitor continuously monitors user activity for a threat  
to the computer system 91. It continuously monitors the user command entries 92, the  
25           system process accounting records 93, and commands entered by the user as stored in  
the user's command history file 94. It compares the command entries 92, system process  
accounting records 93 and commands in the user's command history file to known threat  
events and known attack patterns indicating a computer intrusion 95. If a match occurs 96,  
information and notification is sent to the control function 97. In either case the continuous  
20           session monitoring process continues its dynamic monitoring at step 91.

30           Turning now to Fig. 8, a flow diagram of the control function is shown. The  
controller 125 receives information about events and receives signature information to  
identify the user and type of event 126. Because the controller may be local to the system,  
the system can function in real time for suspicious events. In addition, if the controller is  
35           local, the intrusion detection system can be located entirely within the local host computer.  
The controller then determines the action to be taken and takes appropriate action 127.  
The action may be to log the event to the local system log 128, log the event to a remote  
40           system log 129, disable the user's account 130, block access to the attacking host system  
address 131, trigger a user defined event 132, drop the route to the offending system 133,  
30           block network access from the offending system 134, notify the system administrator 135,  
to ignore the event 136 or any combination of these actions. If the controller is a local 137,  
45           the information can be sent to a local system controller 138. If the controller is not local  
137, the information can be sent to the central system controller 139, which then takes the  
appropriate actions (127-136) instead of the local controller.

35           Fig. 9 shows a block diagram of an alternate embodiment of a host based intrusion  
detection system having a central system controller. The central system controller 150

5 may be part of a network that contains multiple host computers (1 through N) 151-153.  
Each host 151-153 comprises a local controller that sends information about log auditing,  
login anomaly detection, logout anomaly detection, session monitoring and port scan  
detector functions to the central controller. The central controller can perform centralized  
10 5 auditing of events 154, data analysis 155, cross correlation of intrusion activity throughout  
the network 156 and can alert the network system administrator 157 if anomalous activity is  
found. In addition, the central controller 150 can send information about anomalous activity  
found within the system back to the multiple hosts 151-153 so as to alert the hosts.

15 Fig. 10 is a flow diagram of a program set up for the intrusion detection system.  
10 10 Prior to initialization of the intrusion detection system, the system administrator 161 may  
select program functions to run in the intrusion detection system. For example, the system  
administrator may select the log auditing function 162, login anomaly detection 163, logout  
anomaly detection 164, session monitor 165 and port scan detector 166. The system  
20 20 administrator may also select the actions to be taken by the control function if an  
unauthorized user or event occurs 167. If the system administrator chooses not to select  
functions, the preprogrammed default functions will run. If the system administrator  
chooses not to select the actions to be taken by the control function or only changes some  
25 25 of the actions, the preprogrammed default actions will be used. The system administrator may  
also alter the alarm thresholds or use preprogrammed alarm thresholds 168. The system  
administrator may select whether a warning is to be displayed on the system administrators  
graphical user interface 169. The system administrator may also select whether a local or  
30 30 central controller will be used for reporting and for taking action 170.

Fig. 11 is a block diagram of the software modules of the Login Anomaly Detector.  
The Login Anomaly Detector 3 comprises a login audit module 180, first login warning  
35 35 module 181, a foreign domain warning module 182, a multiple concurrent logins module  
183 and an odd login time module 184.

40 The login audit module 180 logs all user logins into the target host computer. This  
information is recorded in the system audit records and the intrusion detection system also  
records this information. The login audit module 180 provides secondary audit trail of user  
activity in case the system audit files are damaged or altered.

45 The first login warning module 181 notifies the control function and/or  
administrators whenever a user logs into a host for the first time. After the first login the  
module will no longer activate. The first login warning module 181 detects a first time login  
by noting whether the user has more than one login in the dynamic user database. It is  
35 35 used to spot users who are not authorized to connect to the computer system.

5                   The foreign domain warning module 182 notifies administrators whenever a  
"foreign" domain login is detected by a user. The foreign domain warning module 182  
checks the Internet domain the user is logging in from as indicated in their user database  
login record. If the domain is not listed in an "allowed" file, it assumes the domain is  
5                   10                  "foreign" and notifies the administrator. The foreign domain warning module 182 allows an  
administrator to spot logins from odd places that are not allowed to connect to the target  
host computer.

15                  The multiple concurrent logins module 183 watches user logins and looks for  
concurrent logins from multiple domains. When a user is logged in from multiple domains  
10                  at the same time, this may be an indication of suspicious activity. The multiple concurrent  
logins module 183 module will monitor logins and if it spots a user that is logged in more  
than once or is logged in from two or more separate domains/hosts, it will notify a control  
function and or the system administrator. Hackers often will log in from multiple hosts  
20                  across the Internet or within a local network. The multiple concurrent logins module 183  
attempts to detect this condition.

25                  The odd login time module 184 monitors user logins and attempts to spot "unusual"  
login times based on past data collected for this user. Odd login times are one of the  
primary indicators of unauthorized system intrusion. The odd login time module 184 runs  
only after a predetermined amount of user logins have been collected by the user  
20                  database. This amount defaults to ten logins, but can be adjusted by the user or system  
administrator to begin comparing login times after any amount has passed, although  
sufficient time should be granted to allow accurate profiling. The theory of operation is to  
take the average login hours from the login tracking field for a particular user. This average  
is used to draw conclusions about the user's login habits including the days they log into  
25                  the computer, the times they log into the computer and how long they stay logged into the  
computer.

40                  This data can be obtained because the login stamp for each login tracking entry is  
dynamically maintained by the login monitoring process. Because the database is  
dynamically generated the signature can be built with intelligence to take advantage of this  
30                  fact to reduce false alarms. This relieves the administrator of having to setup predefined  
login profiles for users. This is a great benefit if you have an eclectic user base who work  
strange hours or login from multiple time zones. The values derived are obtained by  
45                  calculating the days/hours/minutes. These values can be combined in a number of ways to  
determine normal patterns of behavior such as days of the week the user is active and for  
35                  how long they use the system at a time. Further pieces of information can be derived such  
as average hours online per day and the hours they are normally working between.

5 Fig. 12 is a block diagram of the software modules of the Logout Anomaly Detector.  
The Logout Anomaly Detector 7 comprises a logout audit module 190, suspicious entries in  
user's home directory 191, generic file exists module 192, history file truncated/ altered  
module 193, suspicious directory name module 14, altered/missing audit record module  
5 195, network process active module 196, suspicious history file commands module 197,  
10 and rhost file exists module 198.

The logout audit module 190 logs all user logouts from the target host computer.  
This information is recorded in the system audit records and the intrusion detection system  
15 also records this information. The logout audit module 190 provides secondary audit trail of  
10 user activity in case the system audit files are damaged or altered.

20 The suspicious entries in user's home directory module 191 checks the for a  
.rhost" file in the user's home directory with a dangerous entry. Dangerous entries include  
wildcard characters. If a wildcard character is found, the suspicious entries in user's home  
directory module 191 alerts the administrator that a dangerous .rhost file exists. Dangerous  
25 entries indicate suspicious activity for most users and may allow the host system to be  
easily compromised by remote attackers.

30 The generic file exists module 192 module checks an administrator-generated list of  
files to see if one or more of them exist in the user's home directory. This module allows an  
administrator to flag certain files for monitoring (password files, etc.) and generate custom  
20 alerts. A file list is used to parse against the user's directory listing. If a matching file name  
is found the event if flagged and the control function and or system administrator alerted.

35 The history file truncated/ altered module 193 module checks a user's command  
history file for alterations or truncations. Hackers often alter the history file to conceal  
activity on a host. The module checks to determine if the history file is truncated to zero  
bytes long, is missing or deleted and if the history file is a "symbolic link" to another file or  
device. If the history file indicates that any of these conditions have occurred, this may that  
unauthorized activity is being hidden on many attacked hosts. When an altered history file  
is found it is reported to central controller and system the administrator may be notified.

40 The suspicious directory name module 194 detects suspicious directory names.  
30 Hackers will often employ odd directory names in order to hide activity on a host. This  
module searches for common directory name hiding tactics. For example, this module will  
check a user's home directory for odd directory names such as: "..", "...", etc. and notify  
the control function and report them to the administrator if one is found. The directory  
names that can be searched for are configurable by the administrator.  
45 35 The altered/missing audit record module 195 checks to determine if an entry for the  
user's session is missing from the systems audit records (such as utmp, wtmp, event logs

5 or the like). Intruders will use commonly available utilities to purge audit records of their tracks. An altered or missing entry may indicate that the user has deleted or altered their system audit record entry to avoid detection. The altered/missing audit record module 195 checks the pertinent system accounting records (utmp, wtmp, event logs or the like) and  
5 ensures that a matching record exists for the session that the user is logging out of. A  
10 missing record indicates that the user may be trying to conceal their activity on the host. This module attempts to analyze the login/logout records of the host to spot this condition and notify the administrator.

15 The network process active module 196 checks to determine if a  
10 user has logged out of the system and left a process running with a listening network. A listening socket on a network host is a program that is offering services to the Internet. Typically a socket is a number in the range of about 1-65535 that a user can connect to using one of the Internet protocols. An example of common socket numbers include: 23 for Telnet services, 25 for mail services and 80 for worldwide web services. If a user executes  
20 15 a rogue program with a listening socket they can connect back to the system from anywhere on the Internet and bypass normal authentication and audit procedures of the target host. Therefore, programs with listening network sockets run by ordinary users are a security threat and should not be allowed on any host. This could indicate that an unauthorized program is running. The network process active module 196 checks the  
25 20 system process table for network socket programs and alerts the controller and system administrator to this fact.

30 The suspicious history file commands module 197 checks to determine if the history files contain commands that could be considered "suspicious". Most Unix systems have the ability to store a "command history" of all logged in users. This command history is used by  
35 25 the interactive shells to allow users to quickly recall commands either manually or with a script. As part of this feature the interactive shells (depending on their type) will write a "history" file in the user's home directory. Usually this file is called ".history" but other variations exist. The basic operation occurs when a user logs out of the host system.  
40 During this time the login file is flushed to the disk and the login monitor process can begin  
30 reading the most current command history. This module then takes the .history file and compares the command contents to a known database of unusual or suspicious commands. Such commands may include commands that indicate the user has viewed the master password list on the system, the user has added a global wildcard to the system which will allow all hosts access to the system, the user has tried to copy the password file  
45 35 list and the user has tried to switch to system administrator credentials. When suspicious commands are spotted, they are flagged and the appropriate actions can be taken by the

50

5 monitor program to alert the controller and system administrator or disable the user account.

The .rhost file exists module 198 checks for the existence of a ".rhost" file in the user's home directory. The .rhost files contains a list of hosts that are trusted by the system

10 5 when a login is detected originating from them. If a user is originating form a host that is listed in the .rhost file, then they are allowed access to the system as long as their account name matches the owner of the .rhost file. If an entry such as "++" is placed in the .rhost file, it signifies a wildcard and any host can log into the system as the owner of the rhost file. A wildcard entry is always suspicious and is virtually always a dangerous modification.

15 10 If an .rhost file exists in the user's directory, and is greater than zero bytes, then the controller is notified and the system administrator may be alerted.

20 Although the present invention has been described in detail with reference to certain preferred embodiments, other embodiments are possible. Therefore, the spirit and scope of the appended claims should not be limited to the description of the preferred

15 15 embodiments herein.

25

30

35

40

45

50

55

**Claims**

5

10

15

20

25

30

35

40

45

50

55

What is claimed is:

- 5 1. A computer implemented method for detecting intruders in a computer system (1), the  
2 method comprising the steps of:  
3 a. detecting an unauthorized user attempting to enter into a computer system (20) by  
4 comparing actions of the user to a dynamically built profile for the user (22), and if  
5 the action is out of range of the user profile, notifying a control function (37);  
6 b. detecting events that indicate an unauthorized entry into the computer system (49,  
7 75, 90) has occurred and if an event occurs that indicates unauthorized entry,  
8 notifying a control function (55, 85, 97); and  
9 c. executing an action (127) by the control function (125).
- 10 1. 2. The method of claim 1 wherein the dynamically built user profile comprises:  
2 a. dynamically constructing a user profile (22) for each computer user when the  
3 computer user first attempts to log into the computer system (24, 36);  
4 b. dynamically updating the user profile for the user for each attempt by the user to log  
5 into the system after the first attempt (25, 36); and  
6 c. updating the user profile when the user logs out of the computer system (50).
- 15 1. 3. The method of claim 1 further comprising dynamically monitoring computer system log  
2 files (10) for events that indicate an unauthorized attempted entry into the computer  
3 system.
- 20 1. 4. The method of claim 3 wherein the dynamically monitoring system log files comprises:  
2 a. comparing the system log files to events to ignore and ignoring the event if the  
3 system log file indicates a match with an event to ignore (14); and  
4 b. comparing the system log files to events known to indicate an unauthorized entry  
5 event into the computer system (12) and notifying a control function about the  
6 unauthorized entry event;  
7 c. executing the action in response to the event by the control function (17).
- 25 1. 5. The method of claim 1 further comprising:  
2 a. dynamically monitoring user actions after the user has logged into a computer  
3 system for unauthorized access by the user to system information (92), and if  
4 unauthorized access event occurs, notifying a control function (97) about the  
5 unauthorized access and automatically executing a specific action in response to  
6 the event by the control function (127); and  
7 b. dynamically monitoring user actions after the user has logged into a computer  
8 system for corruption of system information by the user (56, 59) and if a corruption  
9 of system information occurs, notifying a control function of the corruption of system  
10 information and executing the action in response by the control function (127).

1       6. The method of claim 1 further comprising:

5       2       a. scanning network ports (76) to determine if a user has connected to more than a  
3               selected number of network ports (83);  
4       b. if the user has exceeded the selected number of network ports (83), notifying the  
5               control function and executing an action in response by the control function (85).

10      1       7. The method of claim 6 wherein the selected number of network ports is set by the  
2               system administrator (161).

15      1       8. The method of claim 1 wherein the detecting events (49, 75, 90) that indicate an  
2               unauthorized entry into the computer system comprises:

15      3       a. detecting anomalous events when a user logs out of the computer system (49)  
4               comprising:  
5               i. monitoring a user's file history to determine if the user's file history has been altered  
6               (51);  
20       7       ii. monitoring computer system files to determine if a modification has been made that  
8               indicates an unauthorized intrusion into the computer system (56);  
9               iii. monitoring a user's computer files to determine if a modification has been made  
10               that indicates an unauthorized intrusion into the computer system (59);  
25       11       iv. determining if a program has been left running that should have stopped running  
12               when the user logs out of the computer system (63); and  
13       b. if an anomalous event has been detected:  
14       14       i. notifying the control function about the anomalous event (55); and  
15       15       ii. allowing the control function to take action in response to the anomalous event  
16               (127).

30      1       9. The method of claim 1 wherein the detecting unauthorized users (20) comprises:

35      2       a. if the user has attempted to log in from a computer host that is not allowed access  
3               to the computer system, notifying the control function about the attempted login  
4               (26); and  
5       b. allowing the control function to take action in response (37).

40      1       10. The method of claim 1 wherein the detecting unauthorized users (20) comprises:  
2       a. if the user attempts to log into the computer system and has an active login (28),  
3               checking to determine if the user is allowed to have more than one login active  
4               simultaneously (29), and if not notifying a control function about the attempted login  
5               (37); and  
6       b. executing an action by the control function (127).

45      1       11. The method of claim 2 wherein the dynamically constructed user profile for each  
2               computer user (22) is selected from the group consisting of storing user name, login

3        terminal, time of creation of initial user profile, time of user's first login, time history of  
5        the user's logins, time periods that the user is allowed to log into the system and total  
5        number of logins for the computer user (24).

1        12. The method of claim 11 wherein the user profile is stored in a user profile database  
2        (36).

10      13. The method of claim 11 wherein dynamically updating the user profile for the user (22)  
2        comprises, for each user selecting from the group consisting of entering a current login  
3        time, login terminal, updating a time history of a user's login and incrementing the total  
4        number of logins (24, 25).

15      14. The method of claim 11 wherein the detecting unauthorized users (20) comprises if the  
2        user has attempted to log in at a time different from the time periods (30) that the user  
3        is allowed to log into the computer system, notifying a control function (37) about the  
4        attempted login and allowing the control function to take action in response (127).

20      15. The method of claim 1 further comprising:  
2        a. dynamically constructing a list of active users logged into the computer system (36);  
3        and  
4        b. dynamically updating the list of active users when a user logs into the system and  
5        logs out of the system (50).

25      16. The method of claim 15 wherein the list of active users (36) comprises information  
2        selected from the group consisting of user name, user terminal and time of user login  
3        (24).

30      17. The method of claim 1 wherein the control function (125) comprises:  
2        a. storing information about unauthorized users and events that indicate an  
3        unauthorized entry into the computer system (126);  
4        b. taking action in response to the unauthorized users and events, the action is  
5        selected from the group consisting of:  
6        i. logging the information in a local controller (128);  
7        ii. sending the information to a network controller (139);  
8        iii. disabling the unauthorized user's account (130);  
9        iv. blocking access to the computer system for the user (131);  
10      v. notifying a system administrator (135); and  
11      vi. ignoring the unauthorized user and unauthorized entry (136).

35      18. The method of claim 17 wherein the action taken is defined by the system administrator  
2        prior to initialization of the intrusion detection system (161).

40      19. The method of claim 1 wherein the control function (125) is located in a local computer  
2        (137) where the unauthorized user and unauthorized entry occurred.

50

5        20. The method of claim 19 further comprising the control function (125) in the local  
2        computer sends information about unauthorized users and anomalous events to a  
3        central computer (139) connected to the local computer (137).

10      21. The method of claim 1 wherein the control function (125) is located in a central  
2        computer (139) connected to the local computer (137).

15      22. The method according to claim 21 further comprising multiple local computers (151-  
2        153) connected to the central computer (150).

20      23. The method of claim 21 wherein the control function (125) in the central computer (150)  
2        comprises:  
3        a. performing centralized analysis of unauthorized users and events (154, 155);  
4        b. performing correlation of unauthorized users and events from the multiple local  
5        computers (156);  
6        c. alerting a central computer system administrator (157); and  
7        d. sending the analysis and correlation results to the multiple local computers (151-  
8        153).

25      24. The method of claim 1 further comprising:  
2        a. for each user, continuously monitoring user activity for a threat to the computer  
3        system (90); and  
4        b. the continuously monitoring comprises analyzing user command entries (92) and  
5        comparing the entries to known threat events and known attack patterns (95)  
6        indicating a computer intrusion and if a match occurs (96), notifying the control  
7        function (97) and allowing the control function to take action in response (127).

30      25. The method of claim 24 further comprising continuously monitoring the system process  
2        accounting records (93) and comparing the entries to known threat events and known  
3        attack patterns (95) indicating a computer intrusion and if a match occurs (96), notifying  
4        the control function (97) and allowing the control function to take action in response  
5        (127).

35      26. The method of claim 24 further comprising continuously monitoring commands (94)  
2        entered by the user and comparing the commands to known threat events and known  
3        attack patterns (95) indicating a computer intrusion and if a match occurs (96), notifying  
4        the control function (97) and allowing the control function to take action in response  
5        (127).

40      27. The method of claim 1 further comprising continuously monitoring network port activity  
2        (76) and comparing the activity to known threat events and known attack patterns  
3        indicating a computer intrusion (83, 84) and if a match occurs, notifying the control  
4        function (85) and allowing the control function to take action in response (127).

50

1    28. The method of claim 22 wherein the action taken is selected from the group consisting  
5    2    of logging the event (128, 129), disabling a user account (130), blocking access to the  
3    system (131), initiating a user defined action (132), dropping a route to an attacking  
4    system (133), dropping a route to an attacking user (133), blocking access from an  
5    offending system (134), notifying a system administrator (135) and ignoring the event  
10    6    (136).

1    29. The method as in any of claims 1, 4-6, 8, 9, 14, 17, 24-27 wherein the action comprises  
2    a user specified action (167).

1    30. The method of claim 29 wherein the user specified action is entered by a system  
15    2    administrator (161).

1    31. The method as in any of claims 1, 4-6, 8, 9, 14, 17, 24-27 wherein the action is  
2    automatically executed by the control function (127).

1    32. Computer executable software code stored on a computer readable medium  
20    2    incorporating the method as recited in any of claims 1 through 31.

25

30

35

40

45

50

22

55

1/11

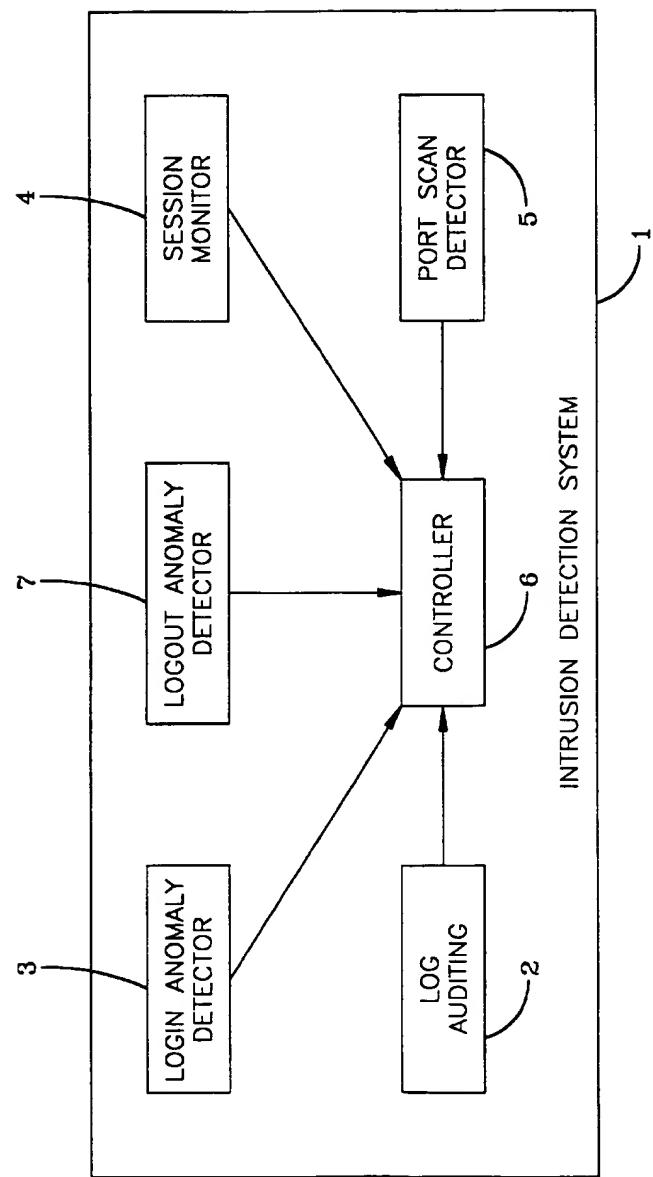
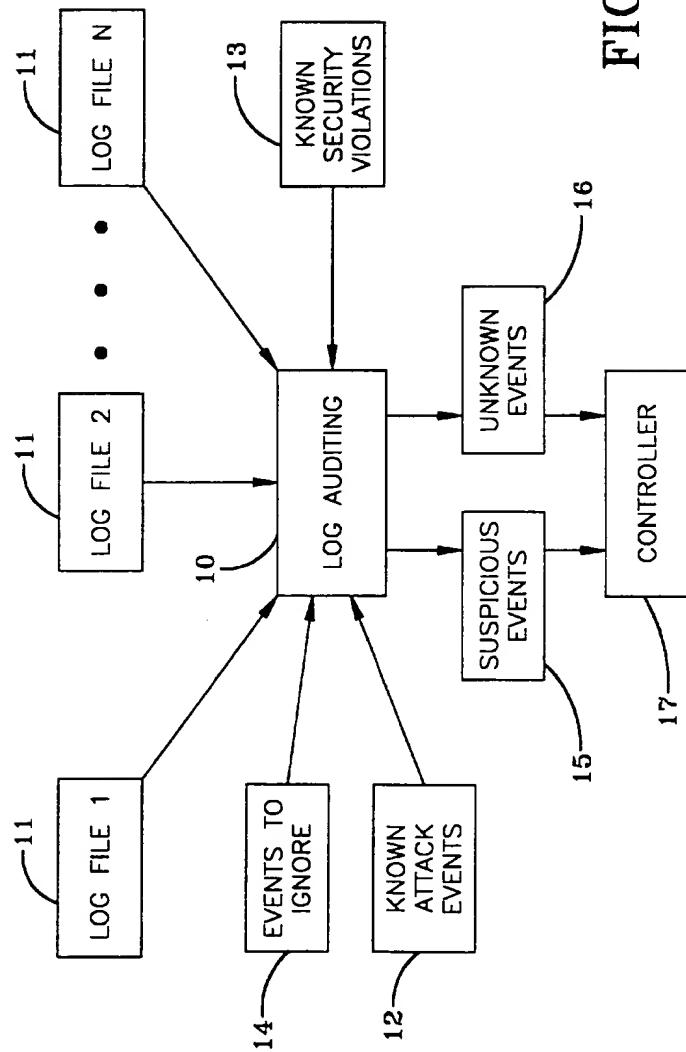


FIG-1

2/11

FIG-2



3/11

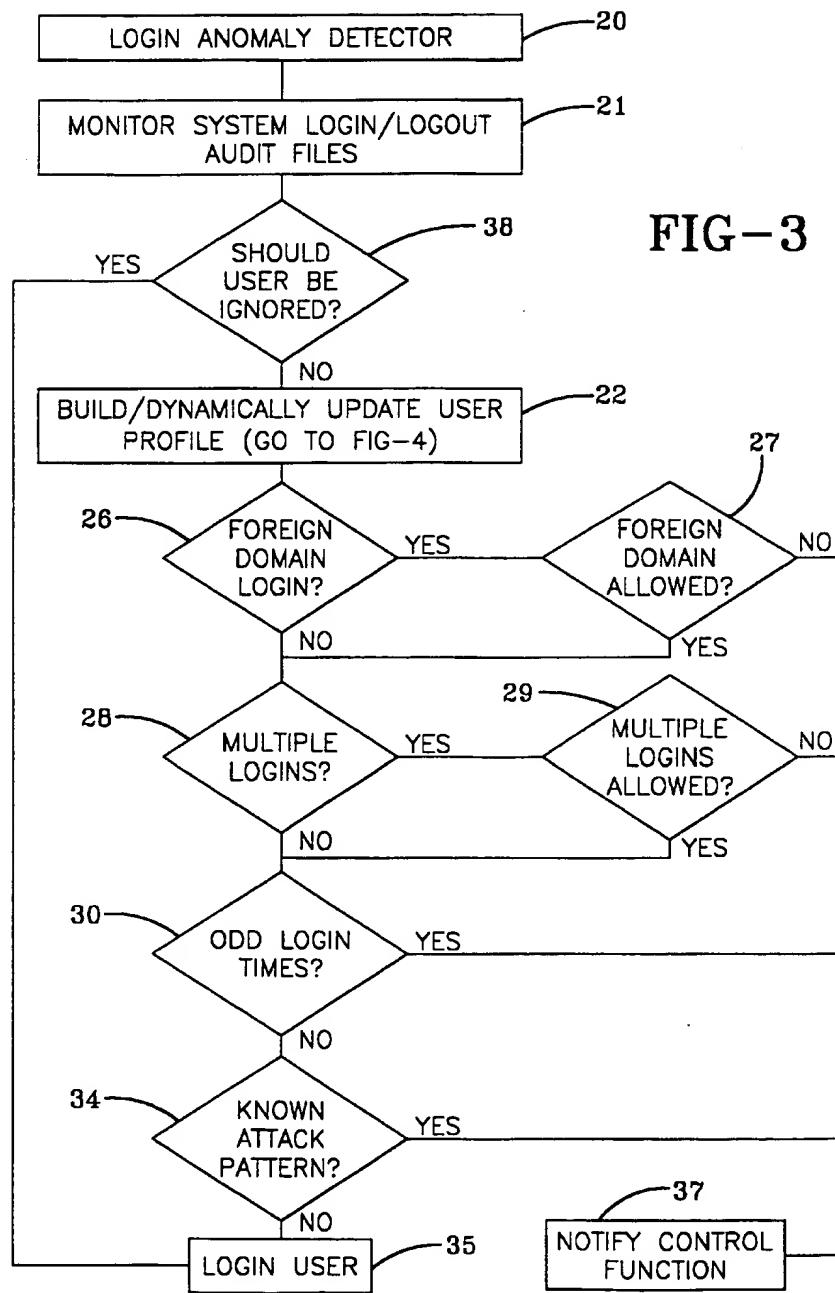
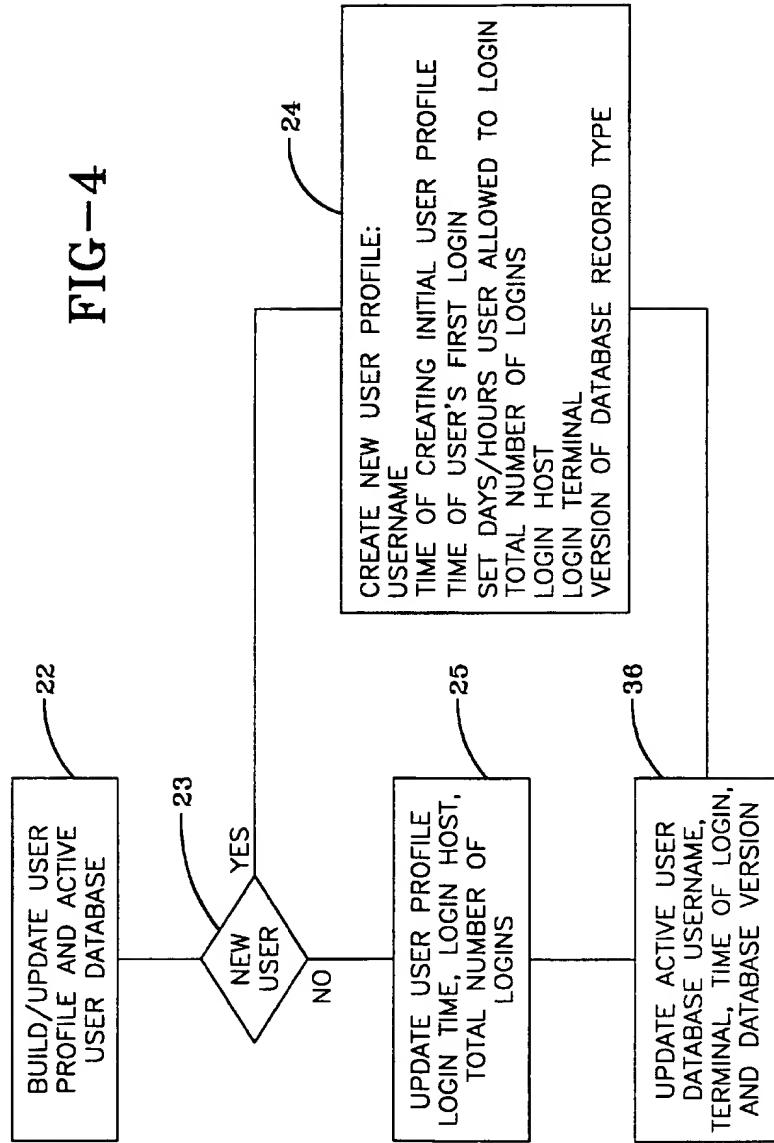


FIG-3

4/11

FIG-4



5/11

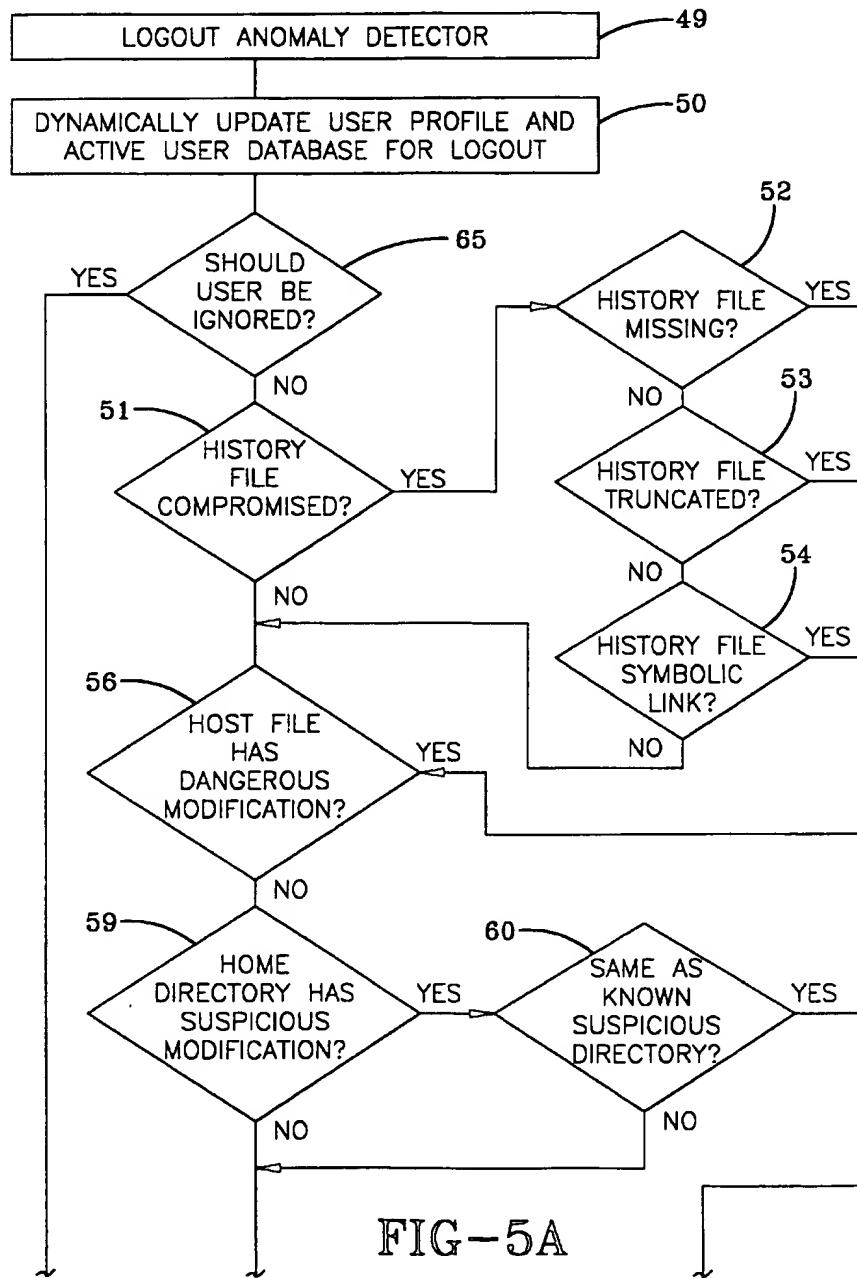


FIG-5A

6/11

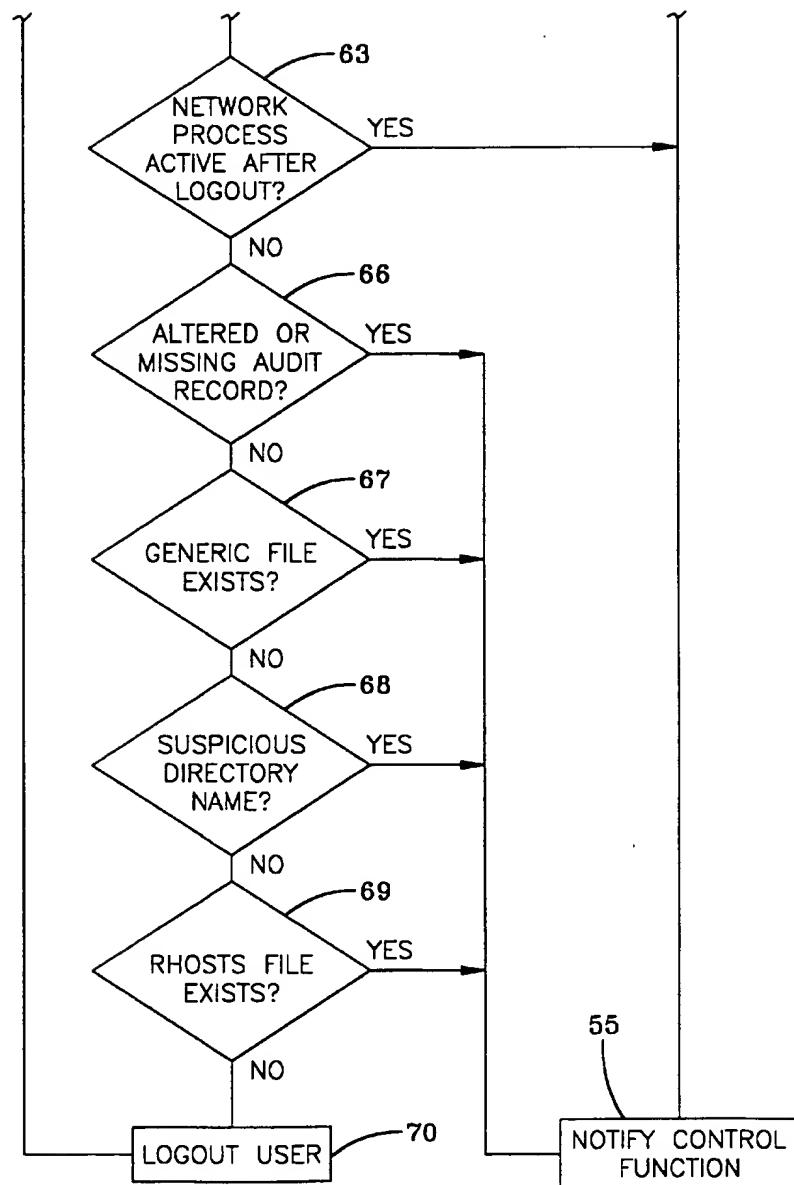
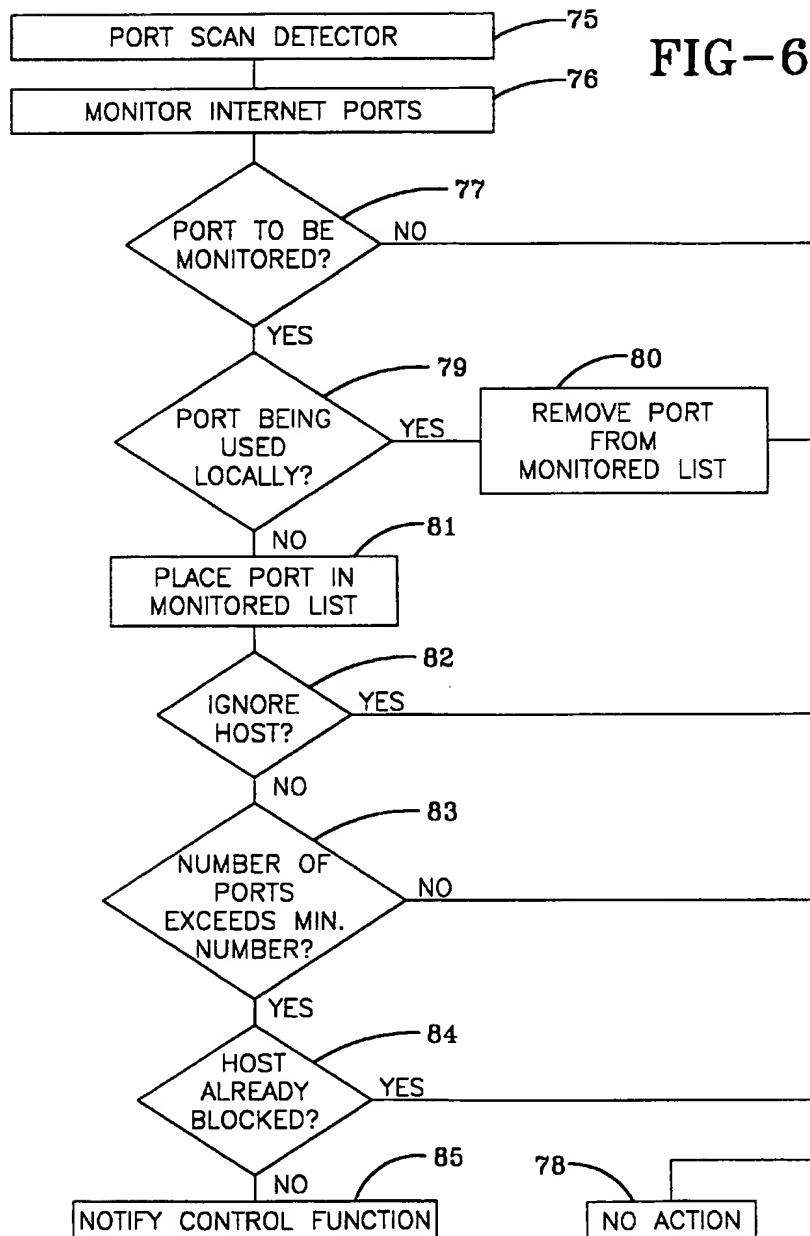


FIG-5B

7/11



8/11

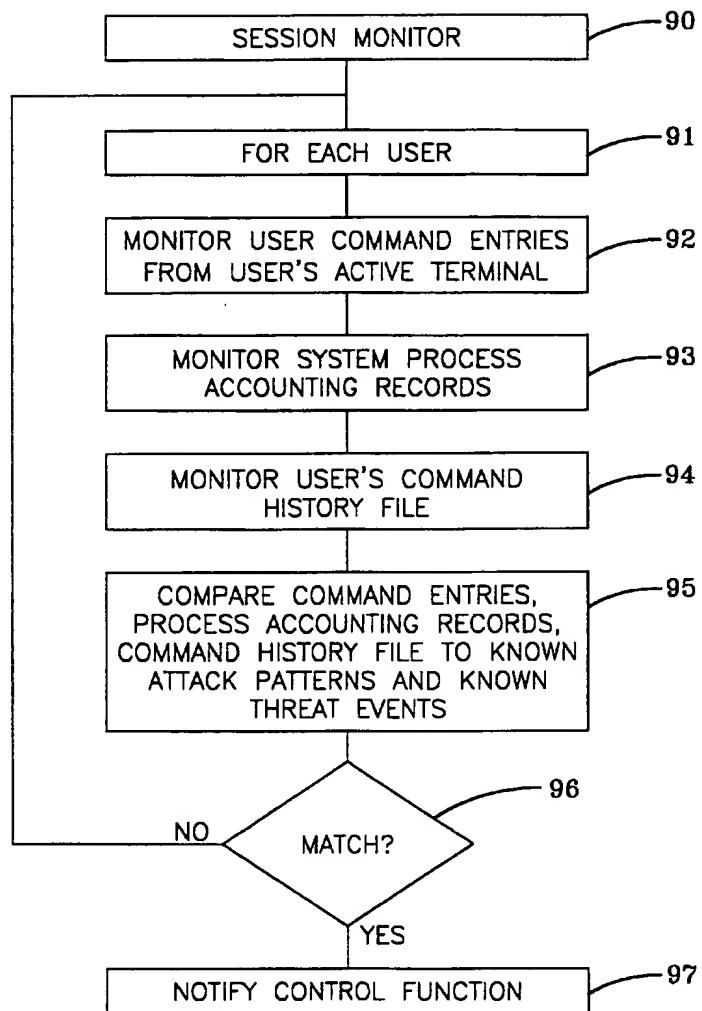


FIG-7

9/11

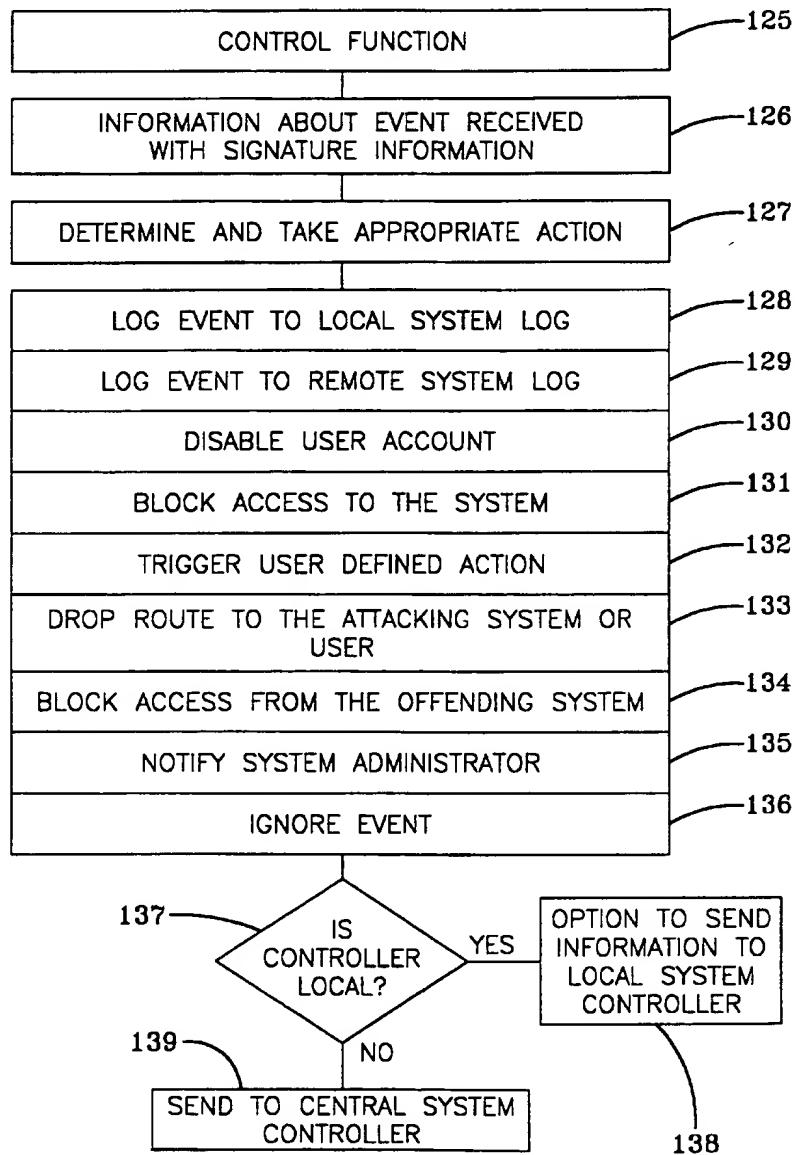
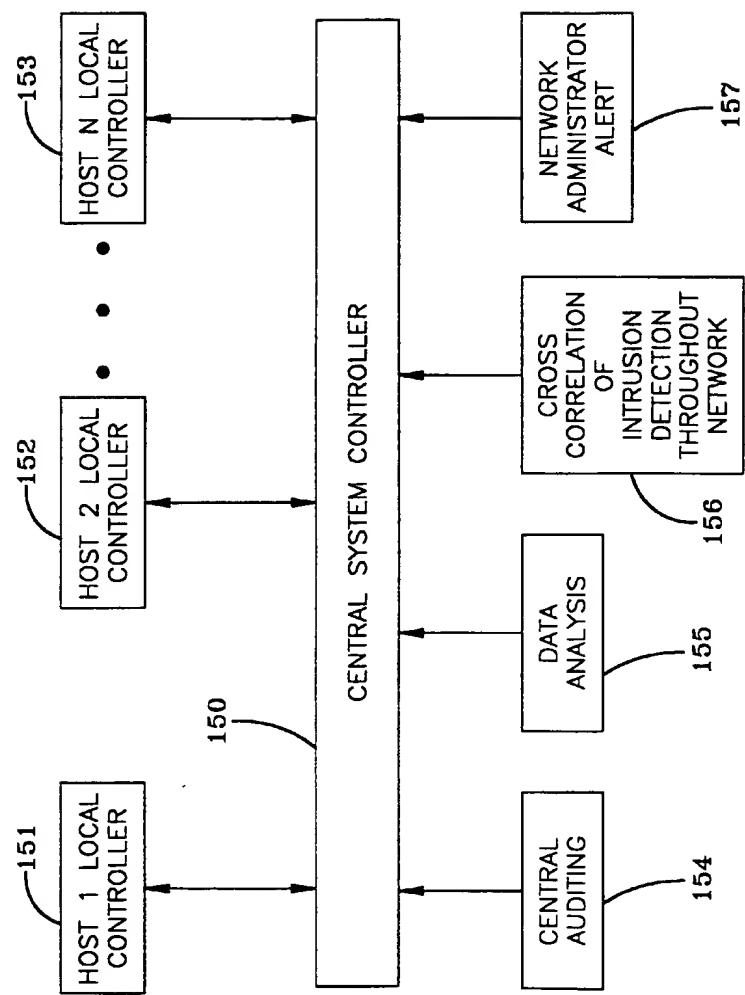


FIG-8

10/11



FIG—9

11/11

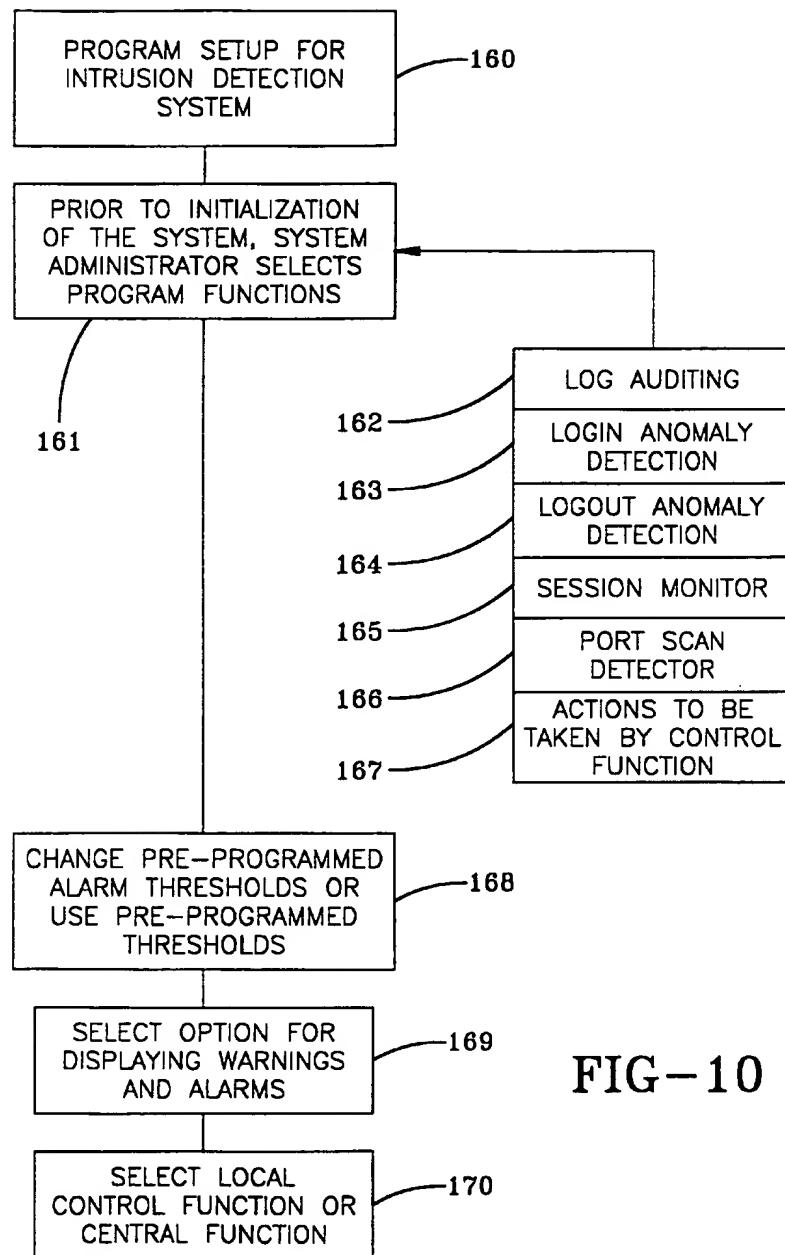


FIG-10

## INTERNATIONAL SEARCH REPORT

International application No. PCT/US00/06313
---

<b>A. CLASSIFICATION OF SUBJECT MATTER</b>		
IPC(7) : H04L 9/32 US CL : 713/200, 164; 709/229 According to International Patent Classification (IPC) or to both national classification and IPC		
<b>B. FIELDS SEARCHED</b>		
Minimum documentation searched (classification system followed by classification symbols)  U.S. : 713/200, 164; 709/229		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)		
<b>C. DOCUMENTS CONSIDERED TO BE RELEVANT</b>		
Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,825,750 A (THOMPSON) 20 October 1998, abstract, column 2 lines 14-19, column 5 lines 14-26, figure 3, column 4 lines 54-67, column 1 lines 47-53, column 4 lines 23-67, column 6 lines 36-51	1-9, 11-19, 24-32
Y,P	US 5,991,881 A (CONKLIN et al) 23 November 1999, abstract, column 1 lines 50-65, figure 6, column 4 lines 9-43, column 5 lines 23-24, column 6 lines 20 and 28-33	1-9, 11-19, 24-32
X,E	US 6,070,244 A (ORCHIER et al) 30 May 2000, abstract, column 12 lines 51-59	10, 20-23
A,B	US 6,073,240 A (KURTZBERG et al) 06 June 2000, disclosure	1-32
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C. <input type="checkbox"/> See patent family annex.		
* Special categories of cited documents: "A" document defining the general state of the art which is not considered to be of particular relevance "B" earlier document published on or after the international filing date "C" document which may throw doubt on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "D" document referring to an oral disclosure, use, exhibition or other means "E" document published prior to the international filing date but later than the priority date claimed		** later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "A" document member of the same patent family
Date of the actual completion of the international search  14 JUNE 2000		Date of mailing of the international search report  03 AUG 2000
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230		Authorized officer <b>GAIL HAYES</b> <i>Joni Hill</i> Telephone No. (703) 305-3900

## INTERNATIONAL SEARCH REPORT

International application No.  
PCT/US00/06313

## C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A.	US 5,495,521 A (RANGACHER) 27 February 1996, disclosure	1-32
A,E	US 6,044,465 A (DUTCHER et al) 28 March 2000, disclosure	1-32

Form PCT/ISA/210 (continuation of second sheet) (July 1998)\*